

RTU studiju kurss "Programmatūras izstrādes fundamentālie aspekti"

33000 Datorzinātnes, informācijas tehnoloģijas un enerģētikas fakultāte

Vispārējā informācija

Kods	DE0163
Nosaukums	Programmatūras izstrādes fundamentālie aspekti
Studiju kursa statuss programmā	Obligātais/Ierobežotās izvēles
Atbildīgais mācītbspēks	Ērika Nazaruka - Doktors, Asociētais profesors
Apjoms daļās un kredītpunktos	1 daļa, 8.0 kredītpunkti
Studiju kursa īstenošanas valodas	LV, EN
Anotācija	Studiju kursa ietvaros studentiem ir iespēja padziļināti ieskatīties programmatūras izstrādes fundamentālajos aspektos, sākot no objektorientētās programmatūras izstrādes matemātiskās formalizācijas un beidzot ar matemātiskajos modeļos sakņotās programmatūras izstrādes formalizācijas perspektīvām. Studentiem ir piedāvāts apgūt topoloģiskās modelēšanas attīstības vēsturi un matemātiskos pamatus, matemātiskā topoloģiskā modeļa pielietošanu programmatūras izstrādē funkcionēšanas analīzei un sintēzei, projektēšanas un analīzes modeļu verifikācijai, prasību kvalitātes nodrošināšanai, sistēmas struktūras un uzvedības noteikšanai. Papildus ir piedāvāts izpētīt topoloģisko modeļu pielietošanu medicīnā un hibrīdu sistēmu pētīšanā.
Mērķis un uzdevumi, izteikti kompetencēs un prasmēs	Studiju kursa mērķis ir sniegt zināšanas programmatūras izstrādes matemātiskās formalizācijas pieejās sākot no programmēšanas valodām un beidzot ar izstrādes procesiem. Studiju kursa uzdevumi: 1. Attīstīt studentu kompetenci matemātisko modeļu pielietošanā sistēmas analīzē un projektēšanā un attīstīt prasmes sistēmas projektējuma kvalitātes nodrošināšanā, izmantojot topoloģisko modelēšanu. 2. Pilnveidot studentu prasmes zinātniskajā diskusijā par programmatūras izstrādes fundamentālajiem aspektiem.
Patstāvīgais darbs, tā organizācija un uzdevumi	Patstāvīgais darbs ietver referāta izstrādi, kurā studenti integrē iegūtās zināšanas par programmatūras izstrādes fundamentālajiem aspektiem un iegūst praktisko pieredzi programmatūras izstrādes formalizācijā.
Literatūra	Obligātā / Obligatory: 1. J. Osis, E. Asnina. Model-Driven Domain Analysis and Software Development: Architectures and Functions. IGI Global, Hershey - New York, 2011, 514 p. 2. Osis, J., Doniņš, U. Topological UML Modeling: An Improved Approach for Domain Modeling and Software Development (1st Edition). Amsterdam: Elsevier, 2017. 276 lpp. ISBN 9780128054765. 3. Thomas Stahl, Markus Volter. Model-Driven Software Development: Technology, Engineering, Management. Wiley, 2003. 4. Anneke Kleppe, Jos Warmer, Wim Bast. MDA Explained: The Model Driven Architecture(TM): Practice and Promise. Addison-Wesley Professional, 2003 5. Osis, J., Grundspenķis, J., Markovičs, Z. Sarežģītu heterogēnu sistēmu topoloģiskā modelēšana: teorija un lietojumi. Rīga: RTU, 2012. 407 lpp. ISBN 978-9934-507-01-4. 6. Mistrik I, Brown AW, Ali Babar M. Agile Software Architecture: Aligning Agile Processes and Software Architectures. Morgan Kaufmann; 2014. Database: eBook Academic Collection (EBSCOhost). 7. Mitchell JC. Concepts in Programming Languages. Cambridge University Press; 2003. Database: eBook Academic Collection (EBSCOhost). Papildu / Additional: 1. J.J.Osis. Topologiskā modelēšana sistēmās. ABT, ANlatvSSR, 1969. 2. J.Osis, L.Beghi. Topological Modelling of Biological Systems. Elsevier Science Publishing, Oxford, UK, 1997, 337-342. 3. Patterson DA, Hennessy JL, Ashenden PJ. Computer Organization and Design, Revised Printing, Third Edition: The Hardware/Software Interface. Vol 3rd ed. rev. Morgan Kaufmann; 2007. Database: eBook Academic Collection (EBSCOhost). 4. ACAA. Ada 202x Language Reference Manual. 2020. URL: http://www.ada-auth.org/standards/ada2x.html 5. University of Oslo Library. Ole-Johan Dahl, Kristen Nygaard and SIMULA. URL: https://www.ub.uio.no/english/subjects/naturalscience-technology/informatics/themes/dns/index.html
Nepieciešamās priekšzināšanas	Programmatūras izstrādes pamati, objektorientētās programmatūras izstrādes pamati.

Studiju kursa saturs

Saturs	Pilna un nepilna laika klātienē studijas		Nepilna laika neklātienē studijas	
	Kontakt stundas	Patstāv. darbs	Kontakt stundas	Patstāv. darbs
Objektorientētās paradigmas vēsturiskās attīstības atspoguļojums programmēšanas valodās.	2	8	0	0
Objektorientētās programmatūras izstrādes dzīves cikli. Spējās metodes.	4	10	0	0
Modelēšanas valodas un to formalizācijas pakāpe. Meta-modelēšana.	6	2	0	0
Modeļos sakņotās izstrādes dzīvescikls un formalizācijas pakāpe.	2	2	0	0

Meta-modeļi un meta-modelēšana kā formalizācijas bāze.	4	2	0	0
Programmatūras izstrādes agrīnās stadijas un formalizācijas tendences.	6	2	0	0
Topoloģiskās modelēšanas attīstības vēsture.	2	2	0	0
Topoloģiskās modelēšanas matemātiskā bāze.	4	4	0	0
Topoloģiskās modelēšanas meta-līmeņi un meta-modeļi.	4	2	0	0
Topoloģiskā modelēšana sistēmu biznesa procesu, objektu un likumu noteikšanai.	6	2	0	0
Topoloģiskā modelēšana sistēmas un programmatūras prasību noteikšanai un izsekojamībai, modeļu verifikācijai.	8	8	0	0
Topoloģiskie modeļi sistēmas struktūras un uzvedības noteikšanai.	8	8	0	0
Topoloģiskie modeļi sistēmas modeļu izpildei objektorientētā analizē un projektēšanā.	8	8	0	0
Topoloģiskajā modelī atspoguļotu zināšanu izplatīšana programmatūras izstrādes artefaktos.	4	4	0	0
Topoloģiskie modeļi programmatūras pirmkoda ģenerācijai.	6	10	0	0
Topoloģisko modeļu pielietošanas iespējas medicīnā un hibrīdsistēmu pētīšanā.	2	0	0	0
Patstāvīgā darba izstrāde.	0	40	0	0
Patstāvīgā darba rezultātu prezentācija.	4	0	0	0
Konsultācijas	2	0	0	0
Eksāmens	4	0	0	0
Kopā:	86	114	0	0

Sasniedzamie studiju rezultāti un to vērtēšana

Sasniedzamie studiju rezultāti	Rezultātu vērtēšanas metodes
Spēj diskutēt par objektorientētu programmēšanas valodu vēsturisko attīstību un modernām tendencēm.	Sekmīgi novērtēts eksāmens. Kritēriji: spēj sistematizēt OOP valodu attīstības tendences; spēj secināt par OOP valodu attīstības iemesliem; spēj novērtēt OOP valodu attīstības tendences.
Spēj salīdzināt un izskaidrot atšķirības un līdzības modernās OO un modeļos sakņotās programmatūras izstrādes metodēs un pieejās.	Sekmīgi novērtēts eksāmens. Kritēriji: spēj izskaidrot modernās metodes un pieejas; spēj secināt par moderno metožu un pieeju stiprajām un vājajām pusēm; spēj salīdzināt modernās metodes un pieejas uzdotajā kontekstā.
Spēj izvērtēt modeļu, metožu un pieeju matemātiskās formalizācijas pakāpi, nepieciešamību un līdzekļus.	Sekmīgi novērtēts eksāmens. Kritēriji: spēj paskaidrot kas ir matemātiskā formalizācija; spēj novērtēt objektu formalizācijas pakāpi; spēj izlemt par formalizācijas nepieciešamību un līdzekļiem.
Spēj sistematiski un formālā veidā risināt sarežģītas datorzinātnes un informācijas tehnoloģijas problēmas.	Sekmīgi aizstāvēts patstāvīgais darbs. Kritēriji: spēj demonstrēt problēmas tipisko risinājumu; spēj modificēt (vai kombinēt) problēmas tipisko risinājumu atbilstoši uzdotajai situācijai; spēj argumentēti pamatot savu izvēli.
Spēj integrēt savas zināšanas un pieredzi ar programmatūras izstrādes formalizācijas tendencēm, novērtēt to esošo stāvokli, kā arī pastāvošus riskus.	Sekmīgi aizstāvēts patstāvīgais darbs. Kritēriji: spēj paskaidrot zināmas formalizācijas tendences; spēj novērtēt mūsdienu formalizācijas tendenču vājas un stiprās puses; spēj novērtēt saistītus ar formalizāciju riskus.
Spēj īstenot zinātniskus un praktiskus eksperimentus un apstrādāt liela apjoma datus, ieskaitot atvērto datu radišanu.	Sekmīgi aizstāvēts patstāvīgais darbs. Kritēriji: spēj novērtēt un analizēt liela apjoma datus; spēj plānot un veikt eksperimentus un apstrādāt eksperimentu rezultātus.
Spēj novērtēt programmatūras izstrādes formalizācijas attīstības tendences un izstrādāt jaunus metodes un algoritmus, kuri uzlabotu sarežģītu sistēmu modelēšanu, projektēšanu un izstrādi.	Sekmīgi aizstāvēts patstāvīgais darbs. Kritēriji: spēj salīdzināt formalizācijas attīstības tendences; spēj integrēt zināšanas no dažādām zinātnes apgabaliem jaunu formalizētu metožu un algoritmu izstrādē.

Studiju rezultātu vērtēšanas kritēriji

Kritērijs	% no kopējā vērtējuma
Eksāmens (obligāta, sekmīgi nokārtota daļa)	40
Patstāvīgā darba (referāta) ietvaros paveiktā pētījuma apjoms un atbilstība uzdevumam (obligāta, sekmīgi novērtēta daļa)	40
Patstāvīgā darba (referāta) ietvaros iegūto rezultātu prezentācija un diskusija (obligāta, sekmīgi nokārtota daļa)	20
Kopā:	100

Studiju kursa plānojums

Daļa	KP	Stundas			Pārbaudījumi		
		Lekcijas	Prakt d.	Laborat	Ieskaite	Eksām.	Darbs
1.	8.0	32.0	32.0	16.0		*	